103. (Amended) The system of claim 99, further including:

means for providing to the first computing environment the stub class instance during runtime operations.

104. (Amended) The system of claim 99, wherein the stub class instance is included in a second computing environment.

REMARKS

In the Final Office Action dated January 31, 2003, the Examiner rejected claims 72-77 and 104 under 35 U.S.C. § 112, second paragraph; rejected claims 34, 39-41, 43, 53, 54, 59-61, and 63 under 35 U.S.C. § 102(a) as being anticipated by Gaines (U.S. Patent No. 5,961,582); rejected claims 35-38 and 55-58 under 35 U.S.C. § 103(a) as being unpatentable over Gaines in view of Priven et al. (U.S. Patent No. 5,327,559); rejected claims 64, 66, 68-71, 73, 75-78, 80, 82-85, 87, 89-92, 94, 96-99, 101, and 103-105 under 35 U.S.C. § 103(a) as being unpatentable over Gaines in view of Hill et al. (U.S. Patent No. 5,511,197); and rejected claims 65, 67, 72, 74, 79, 81, 86, 88, 93, 95, 100, and 102 under 35 U.S.C. § 103(a) as being unpatentable over Gaines in view of Hill et al. and further in view of Priven et al.

By this amendment, Applicants propose to cancel claims 43, 63, 69, 76, 83, 90, and 97 without prejudice or disclaimer and amend claims 34, 53, 54, 64, 67, 68, 71-75, 77, 78, 82, 85, 89, 92, 96, 99, 103, and 104. Based on the proposed amendments and the following remarks, Applicants respectfully traverse the rejections of claims 34-41,

-9-

43-61, and 63-105 under 35 U.S.C. §§ 102(a) and 103(a) and request the timely allowance of the pending claims.

I.   Withdrawn Claims 44-52

Regarding the non-elected claims 44-52, Applicants are not required to cancel these claims at this stage of prosecution of this application (see M.P.E.P. § 821). For example, Applicants have a right to conform the non-elected claims to embrace an allowed generic claim (see M.P.E.P. § 821.02). Accordingly, Applicants will consider canceling and/or amending non-elected claims 44-52 upon receiving a Notice of Allowance for this application.

II.   The Rejection of Claims 72-77 and 104 under 35 U.S.C. § 112, second paragraph

Regarding claims 72-77 and 104, Applicants propose to amend these claims in the manner suggested by the Examiner. In particular, Applicants propose to change the dependencies of claims 72-77 from claim 70 to independent claim 71. Further, Applicants propose to change "the second computing environment" in claim 104 to "a second computing environment." Accordingly, Applicants respectfully request that the rejection of these claims under § 112, second paragraph be withdrawn. Since the Examiner examined these claims as if they were drafted in accordance with the amendment herein, Applicants submit that the proposed amendments to claims 72-77 and 104 have not been made for reasons of patentability.

III. The Rejection of Claims 34, 39-41, 43, 53, 54, 59-61, and 63 under 35 U.S.C. § 102(a)

Applicants respectfully traverse the rejections of claims 34, 39-41, 43, 53, 54, 59-61, and 63 under 35 U.S.C. § 102(a) because <u>Gaines</u> does not teach every recitation of these claims.

<u>Gaines</u> teaches a system for providing access to remote resources in a distributed computing environment. The system includes one or more computers that include a virtual operating system that is capable of executing programs in a host-independent manner. The system allows a program to be executed on a local and remote host computer using the virtual operating system operating within each computer. In one embodiment, <u>Gaines</u> discloses a system where a transferable program 302 comprises elements of a GUI 203 that is provided from a remote computer to a server computer (see <u>Gaines</u>, col. 14, lines 15-27).

In contrast, claim 34 recites a combination of steps including, among other steps, sending a portion of the code to a second program, where the portion of the code is based on stub code obtained from the second abstract computing machine. <u>Gaines</u> does not teach at least this operation. Instead, <u>Gaines</u> discloses virtual applications that are used by the virtual operating systems to access and/or execute resources within a host computer (see <u>Gaines</u>, col. 6, lines 22-36). The transferable program 302 and elements of the GUI 203 included in a first computer disclosed by <u>Gaines</u> are not based on any information provided by a second computer receiving transferable program 302 and interface elements included in the program, but rather are program code hosted at the first computer. Accordingly, <u>Gaines</u> cannot teach at least sending a

portion of the code to a second program, where the portion of the code is based on stub code obtained from the second abstract computing machine, as recited in claim 34. Because <u>Gaines</u> fails to teach every recitation of claim 34, Applicants request that the rejection of claim 34 under 35 U.S.C. § 102(a) be withdrawn and the claim allowed.

Claims 39-41 depend from claim 34. As explained, claim 34 is distinguishable from <u>Gaines</u>. Accordingly, claims 39-41 are also distinguishable from this reference for at least the same reasons set forth for claim 34, and Applicants respectfully request that the rejection of these claims under 35 U.S.C. § 102(a) be withdrawn and the claims allowed.

Claims 53 and 54 include recitations similar to claim 34. As explained, claim 34 is distinguishable from <u>Gaines</u>. Accordingly, claims 53 and 54 are also distinguishable from this reference for at least the same reasons in connection with claim 34, and Applicants respectfully request that the rejection of these claims under 35 U.S.C. § 102(a) be withdrawn and the claims allowed.

Claims 59-61 depend from claim 54. As explained, claim 54 is distinguishable from <u>Gaines</u>. Accordingly, claims 59-61 are also distinguishable from this reference for at least the same reasons in connection with claim 54, and Applicants respectfully request that the rejection of these claims under 35 U.S.C. § 102(a) be withdrawn and the claims allowed.

IV.   The Rejection of Claims 35-38 and 55-58 under 35 U.S.C. § 103(a)

Regarding claims 35-38 and 55-58, these claims depend from claims 34 and 54, respectively. As explained, claims 34 and 54 are distinguishable from Gaines. Accordingly, claims 35-38 and 55-58 are also distinguishable from this reference for at least the same reasons in connection with claims 34 and 54. Further, Priven et al. is not sufficient to overcome the deficiencies of Gaines. Because Gaines and Priven et al., alone or in combination, fail to teach or suggest the recitations of claims 35-38 and 55-58, Applicants respectfully request that the rejection of these claims under 35 U.S.C. § 103(a) be withdrawn and the claims allowed.

The Rejection of Claims 64-105 under 35 U.S.C. § 103(a)

Regarding claim 64, the Examiner implicitly admits that Gaines does not teach sending a portion of the first code "based on second code obtained from" the second computing environment (see Office Action, page 5, paragraph 8). The Examiner, however, asserts that Hill et al. teaches this recitation and that it would have been obvious to "send [the portion of the first code] based on a second code." Applicants respectfully disagree.

Hill et al. teaches a method for a client machine to access remotely stored data, such as a formula stored in a specific cell of a spreadsheet on a server machine. To do so, the client sends a message to a local spreadsheet proxy, the message including a method name (e.g., GetCell) and a string corresponding to the specific cell (e.g., A1) (see Hill et al., col. 7, line 64 - col. 8, line 10). The local spreadsheet proxy

-13-

communicates with a spreadsheet stub on the server machine that eventually returns an address of a remote cell stub and an unmarshal class identifier (see Hill et al., col. 8, lines 13-20). The remote cell stub contains a pointer to the cell in the spreadsheet. The unmarshal class identifier is used by the local spreadsheet proxy to create a local cell proxy to communicate with the remote cell stub (see Hill et al., col. 8, lines 21-44).

However, Hill et al. does not teach sending a portion of the first code from the first computing environment to the second computing environment based on a stub class instance obtained from the second computing environment, as recited in claim 64 (previously recited in proposed canceled claim 69). As explained, Hill et al. teaches client 402 receiving an identifier from server 401 used to create a proxy that communicates with a remote stub, and not receiving the stub itself.

Accordingly, because Hill et al. and Gaines, alone or in combination, does not teach or suggest a stub class instance that is obtained from a second computing environment, as recited in claim 64, Applicants request that the rejection of this claim under 35 U.S.C. § 103(a) be withdrawn and the claim allowed.

Claims 65-68 and 70 depend from claim 64. As explained, claim 64 is distinguishable from Gaines and Hill et al. Accordingly, claims 65-68 and 70 are also distinguishable from these references for at least the same reasons in connection with claim 64. Therefore, Applicants respectfully request that the rejection of claims 65-68 and 70 under 35 U.S.C. § 103(a) be withdrawn and the claims allowed.

Claims 71, 78, 85, 92, and 99 include recitations similar to claim 64. As explained, claim 64 is distinguishable from Gaines and Hill et al. Accordingly, claims 71, 78, 85, 92, and 99 are also distinguishable from these references for at least the

same reasons in connection with claim 64. Therefore, Applicants respectfully request that the rejection of claims 71, 78, 85, 92, and 99 under 35 U.S.C. § 103(a) be withdrawn and the claims allowed.

Claims 72-75, and 77; 79-82, and 84; 86-89, and 91; 93-96, and 98; and 100-105 depend from claims 71, 78, 85, 92, and 99, respectively. As explained, claims 71, 78, 85, 92, and 99 are distinguishable from Gaines and Hill et al. Accordingly, claims 72-75, and 77, 79-82, and 84, 86-89, and 91, 93-96, and 98, and 100-105 are also distinguishable from these references for at least the same reasons in connection with claims 71, 78, 85, 92, and 99. Further, with respect to claims 75, 67, 72, 74, 79, 81, 86, 88, 93, 95, 100, and 102, Priven et al. fails to teach or suggest sending or receiving a portion of a first code based on a stub class instance obtained from or provided by a second computing environment, as recited in these claims. Therefore, Applicants respectfully request that the rejection of claims 72-75, and 77, 79-82, and 84, 86-89, and 91, 93-96, and 98, and 100-105 under 35 U.S.C. § 103(a) be withdrawn and the claims allowed.

Applicants respectfully request that this Amendment under 37 C.F.R. § 1.116 be entered by the Examiner, placing claims 34-41, 53-61, 64-68, 70-75, 77-82, 84-89, 91-96, and 98-105 in condition for allowance. Applicants submit that the proposed amendments of claims 34, 53, 54, 64, 67, 68, 71-75, 77, 78, 82, 85, 89, 92, 96, 99, 103, and 104 do not raise new issues or necessitate the undertaking of any additional search of the art by the Examiner, since all of the elements and their relationships claimed were either earlier claimed or inherent in the claims as examined. Therefore, this Amendment should allow for immediate action by the Examiner.

Furthermore, Applicants respectfully point out that the final action by the Examiner presented some new arguments as to the application of the art against Applicants' invention. It is respectfully submitted that the entering of the Amendment would allow the Applicants to reply to the final rejections and place the application in condition for allowance.

Finally, Applicants submit that the entry of the amendment would place the application in better form for appeal, should the Examiner dispute the patentability of the pending claims.

In view of the foregoing remarks, Applicants submit that this claimed invention, as amended, is neither anticipated nor rendered obvious in view of the prior art references cited against this application. Applicants therefore request the entry of this Amendment, the Examiner's reconsideration and reexamination of the application, and the timely allowance of the pending claims.

Please grant any extensions of time required to enter this response and charge any additional required fees to our deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: March 25, 2003
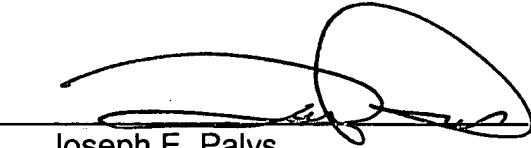
By:_____
Joseph E. Palys
Reg. No. 46,508

-17-

APPENDIX OF CLAIM AMENDMENTS

Please cancel claims 43, 63, 69, 76, 83, 90, and 97 without prejudice or disclaimer and amend claims 34, 53, 54, 64, 67, 68, 71-75, 77, 78, 82, 85, 89, 92, 96, 99, 103, and 104, as follows:

34.   (Amended) A method in a data processing system having a first program containing code and having a second program, the method comprising the steps of:

providing a first abstract computing machine to the data processing system;

providing a second abstract computing machine to the data processing system

running the first program on the first abstract computing machine;

running the second program on the second abstract computing machine;

sending a portion of the code from the first program to the second program, wherein the portion of the code is based on stub code obtained from the second abstract computing machine; and

running the portion of the code by the second program on the second abstract computing machine.

53.   (Amended) A data processing system having a first computing system with a first program containing code and having a second computer system with a second program, comprising:

means for providing a first abstract computing machine to the first computer system;

means for providing a second abstract computing machine to the second computer system;

means for running the first program on the first abstract computing machine;

means for running the second program on the second abstract computing machine;

means for sending a portion of the code from the first program to the second program, wherein the portion of the code is based on stub code obtained from the second abstract computing machine; and

means for running the portion of the code by the second program on the second abstract computing machine in a manner as the code is run on the first abstract computing machine.

54. (Amended) A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system having a first program containing code and having a second program, the method comprising the steps of:

providing a first abstract computing machine to the data processing system;

providing a second abstract computing machine to the data processing system

running the first program on the first abstract computing machine;

running the second program on the second abstract computing machine;

sending a portion of the code from the first program to the second program, wherein the portion of the code is based on stub code obtained from the second abstract computing machine; and

running the portion of the code by the second program on the second abstract computing machine.

64.   (Amended) A method performed in a data processing system including a first computing environment and a second computing environment, the method comprising:

executing a first program including first code on the first computing environment;

sending a portion of the first code from the first computing environment to the second computing environment based on [second code] a stub class instance obtained from the second computing environment;

executing the portion of the first code on the second computing environment; and

returning results of the executed portion of the first code to the first computing environment.

67.   (Amended) The method of claim 64, wherein the portion of the first code is part of an object and executing the portion of the first code includes:

invoking a function included in the second computing environment; and

returning to the first computing environment the object as a result of the invocation.

68.   (Amended) The method of claim 64, wherein the [second code] stub class instance is provided to the first computing environment by the second computing environment during runtime operations.

71.   (Amended) A method performed in a data processing system including a first computing environment and a second computing environment, the method performed by the second computing environment comprising:

receiving a portion of first code included in a program executing in the first computing environment based on [second code] a stub class instance provided to the first computing environment by the second computing environment;

executing the portion of the first code; and

returning results of the executed portion of the first code to the first computing environment.

72.   (Amended) The method of claim [70] 71, wherein receiving a portion of the first code includes:

receiving an object containing the portion of the first code.

73.   (Amended) The method of claim [70] 71, wherein executing the portion of the first code includes:

invoking a function included in the second computing environment based on a parameter included in the portion of the first code.

74.   (Amended) The method of claim [70] 71, wherein the portion of the first code is part of an object and executing the portion of the first code includes:

invoking a function included in the second computing environment; and

returning the object as a result of the invocation.

75.   (Amended) The method of claim [70] 71, further including:

providing to the first computing environment the [second code] stub class

instance during runtime operations.

77.   (Amended) The method of claim [70] 71, wherein returning results of the

executed portion of the first code to the first computing environment includes:

returning the results to the program.

78.   (Amended) A computer-readable medium containing instructions that

perform a method when executed by a processor, the method performed in a data

processing system including a first computing environment and a second computing

environment and comprising:

executing a first program including first code on the first computing environment;

sending at least a portion of the first code from the first computing environment to

the second computing environment based on [second code] a stub class instance

obtained from the second computing environment;

executing at least the portion of the first code on the second computing

environment; and

returning results of the executed portion of the first code to the first computing

environment.

82.   (Amended) The computer-readable medium of claim 78, wherein the [second code] <u>stub class instance</u> is provided to the first computing environment by the second computing environment during runtime operations.

85.   (Amended) A computer-readable medium including instructions for performing a method when executed by a processor, the method performed in a data processing system including a first computing environment and a second computing environment including the resource, and the method performed by the second computing environment comprising:

receiving a portion of first code included in a program executing in the first computing environment based on [second code] <u>a stub class instance</u> provided to the first computing environment by the second computing environment;

executing the portion of the first code; and

returning results of the executed portion of the first code to the first computing environment.

89.   (Amended) The computer-readable medium of claim 85, further including:

providing to the first computing environment the [second code] <u>stub class instance</u> during runtime operations.

92.   (Amended) A system for executing code in a data processing system including a first computing environment and a second computing environment, the system comprising:

means for executing a first program including first code on the first computing environment;

means for sending a portion of the first code from the first computing environment to the second computing environment based on [second code] a stub class instance obtained from the second computing environment;

means for executing the portion of the first code on the second computing environment; and

means for returning results of the executed portion of the first code to the first computing environment.

96.   (Amended) The system of claim 92, wherein the [second code] stub class instance is provided to the first computing environment by the second computing environment during runtime operations.

99.   (Amended) A system for executing code in a data processing system including a first computing environment, the system comprising:

means for receiving a portion of first code included in a program executing in the first computing environment based on [second code] a stub class instance provided to the first computing environment;

means for executing the portion of the first code; and

means for returning results of the executed portion of the first code to the first computing environment.

103.   (Amended) The system of claim 99, further including:

means for providing to the first computing environment the [second code] stub

class instance during runtime operations.

104.   (Amended) The system of claim 99, wherein the [second code] stub class

instance is [an instance of a stub class] included in [the] a second computing

environment.